CLAIMS

Sub a1

What is claimed is:

1. A computer network comprising:

a server system;

a client system, the server system and the client system executing processes to provide security mechanisms for securing traffic communication between the two systems, the processes including key exchange processes executed when the client system is in an operational state;

logic for detecting whether the client system is in operational state;

a storage device at the client system for storing the results of the key exchange processes;

logic for inhibiting the stored results of the key exchange from being updated until a successful execution of another set of key exchange processes between the server system and the client system;

logic for updating the stored results of the key exchange if the execution of the other set of key exchange processes is successful; and

logic for using results stored in the memory to secure the traffic.

2. The computer network of claim 1, wherein the logic for inhibiting the stored results of the key exchange from being updated is embodied in the client system.

3. The computer network of claim 1, wherein the logic inhibiting the stored results of the key exchange from being updated is embodied in the server system.

4. The computer network of claim 1, wherein the state of the server system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

5. The computer network of claim 1, wherein the state of the client system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

6. The computer network of claim 1, further comprising logic for allowing the traffic communication between the server system and the client system to be sent without security.

7. The computer network of claim 1, wherein the client system is a network device.

8. The computer network of claim 1, wherein the storage device is at least one of an Ethernet device, a coprocessor connected to an Ethernet device, and non-volatile storage that is part of an Ethernet device.

9. The computer network of claim 1, wherein the logic inhibiting the stored results of the key exchange from being updated includes:

logic for sending a signal acknowledging the successful execution of another set of key exchange processes; and

logic for sending a signal confirming receipt of the acknowledgement signal.

10. The computer network of claim 1, wherein the server system contains a storage device for storing the results of the key exchange processes.

11. The computer network of claim 1, further comprising logic for switching the server system to a second server system in the computer network if the server system becomes non-operational, the security mechanisms securing traffic communication between the second server system and the client system.

12. A computer [redacted]able medium for use in conjuncti[redacted]th a server system and a client system for providing security mechanisms for securing traffic communication between the server system and client system, the computer readable medium including computer readable instructions encoded thereon for:

detecting whether the client system is in operational state;

executing first key exchange processes between the server system and the client system when the client system enters the operational state;

storing the results of the first key exchange processes into the client system;

inhibiting the stored results from being updated until a successful execution of a second set of key exchange processes between the server system and the client system;

updating the stored results with the results obtained by the second set of key exchange processes if the execution of the second set of key exchange processes is successful; and

using either the stored results or the updated results to secure the traffic depending on whether the second set of key exchange processes is successful.

13. The computer readable medium of claim 12, wherein the state of the server system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

14. The computer readable medium of claim 12, wherein the state of the client system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

15. The computer readable medium of claim 12, further comprising computer readable instruction encoded thereon for allowing the traffic communication between the server system and the client system to be sent without security.

16. The computer readable medium of claim 12, wherein the results of the key exchange processes are stored into at least one of a network device, a coprocessor connected to a network device, and non-volatile storage that is part of a network device.

17. The computer readable medium of claim 12, wherein the instruction for inhibiting the stored results of the key exchange from being updated includes:

sending a signal acknowledging the successful execution of the second set of key exchange processes; and

sending a signal confirming receipt of the acknowledgement signal.

18. The computer readable medium of claim 12, further comprising computer readable instruction encoded thereon for storing the results of the key exchange processes into the server system.

19. The computer readable medium of claim12, further comprising computer readable instruction encoded thereon for switching the server system to a second server system in the computer network if the server system becomes non-operational, the security mechanisms securing traffic communication between the second server system and the client system.

20. A method of providing security mechanisms for securing traffic communication between a server system and a client system, the method comprising:

detecting whether the client system is in operational state;

executing first key exchange processes between the server system and the client system when the client system enters the operational state;

storing the results of the first key exchange processes into the client system;

inhibiting the stored results from being updated until a successful execution of a second set of key exchange processes between the server system and the client system;

updating the stored results with the results obtained from the second set of key exchange processes if the execution of the second set of key exchange processes is successful; and

using either the stored results or the updated results to secure the traffic depending on whether the second set of key exchange processes is successful.

21. The method of claim 20, wherein the state of the server system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

22. The method of claim 20, wherein the state of the client system includes at least one of "OS up," "OS Hung," "Pre-boot," "OS suspend" and "Cold boot."

23. The method of claim 20, further comprising the step of allowing the traffic communication between the server system and the client system to be sent without security.

24. The method of claim 20, wherein the results of the key exchange processes are stored into at least one of a network device, a coprocessor connected to a network device, and non-volatile storage that is part of a network device.

25. The method of claim 20, wherein the step of inhibiting the stored results of the key exchange from being updated includes:

sending a signal acknowledging the successful execution of the second set of key exchange processes; and

sending a signal confirming receipt of the acknowledgement signal.

26. The method of claim 20, further comprising the step of storing the results of the key exchange processes into the server system.

27. The method [redacted] aim 20, further comprising the st[redacted] switching the server system to a second server system in the computer network if the server system becomes non-operational, the security mechanisms securing traffic communication between the second server system and the client system.